

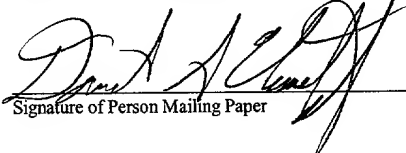
CERTIFICATE OF MAILING
(37 C.F.R. §1.10)

I hereby certify that this paper (along with any referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as "Express Mail Post Office To Addressee" in an envelope addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.

EF322040319US
Express Mail Label No.

August 9, 2001
Date of Deposit

David S. Chametzky
Name of Person Mailing Paper


Signature of Person Mailing Paper

SYSTEM AND METHOD FOR COMPUTER STORAGE SECURITY

BACKGROUND OF THE INVENTION

This invention relates generally to a system and a method for protecting computer storage systems from being accessed by unauthorized computers.

- 5 More particularly, this invention relates to a system and a method that provides for periodic verification of computers sending and receiving data to and from storage devices, storage servers, or storage systems over a computer network.

- Traditionally, computers store data on storage devices, which can be located internally or externally to a computer enclosure. Computers may have access to many storage devices, servers, or systems, some of which are internal and some of which are external. When a storage device is accessed by a computer located within the same enclosure, there is little or no risk of unauthorized access of the stored data by another computer, because the data are not transmitted over a computer network. This configuration provides good security because no other computer can read the data, but does not provide the ability for other computers to directly utilize the same storage device. When a storage device is implemented into a storage area network (SAN), there is an
- 10
- 15

increased risk of unauthorized access of the stored data. The increased risk is caused by the storage device being directly or indirectly connected to many computers over a computer network, such as the Internet, local area networks (LANs), metropolitan area networks (MANs), or wide area networks (WANs).

5 In a storage area network or similar type of network, there will be more than one computer that can have direct access to one or more storage devices. There may also be more than one storage device and the storage device may be controlled by a computer. Because the primary purpose of storage systems is to provide storage, they are typically not equipped with security systems. The
10 computers are connected to the storage systems, storage devices, and servers by a network, which may be large and accessed by many authorized and unauthorized computers. In the case of SANs and other types of networks, there exists a need to provide a security system to prevent unauthorized access of the stored data.

15 Various techniques are available to prevent unauthorized access to computer data. The most common techniques are encryption and authentication protocols. Typically, encryption involves an initiating computer and a servicing computer with a shared secret key and complex algorithms used to encode the data using the shared secret key. Encryption considerably decreases data
20 throughput and increases processing effort. Some encryption protocols required additional hardware to be used, adding to the expense of the system.

Authentication handshaking protocols involve using a shared secret key to establish the communications link between an initiating computer and a servicing

computer. Authentication alone does not provide periodic verification of the initiating computer's identity.

Spoofing is one type of unauthorized access to data in which an invading computer masquerades as the initiating computer after the initiating computer has established a communication link through the authentication protocol with the storage device. The invading computer is able to steal data from the storage device because the invading computer is able to forge its identity as a valid computer.

What is needed is a way to check for unauthorized access of a storage device or system and provides periodic verification of a computer during data transfer between the computer and the storage device or system, while not decreasing data throughput.

SUMMARY OF THE INVENTION

Deficiencies in the prior art are overcome, and an advance in the art is achieved with a method and a system that improves security of a computer network by requiring an initiating computer to periodically reaffirm its identity by transmitting a message, called a "heartbeat" message, to a servicing computer.

The method is carried out in a computer for providing periodic verification of the computer during requests from the computer to a second computer over a communications system. Operationally, the computer establishes an authentication handshake with the second computer and periodically sends messages to the second computer. The requests, which can be requests to send

or receive data, are serviced if the messages are valid and are received within a predetermined time interval. The authentication handshake can include an exchange of a session key and a sequence value. The messages can include the session key and the sequence value, which are processed through a one-way hash function.

In another embodiment, the method is carried out in a computer to provide periodic verification of a second computer during requests from the second computer to the computer over a communications system. Operationally, an authentication handshake is established with the second computer. Messages are periodically received from the second computer and the requests are serviced if the messages are valid and received within a predetermined time interval. The authentication handshake can include an exchange of a session key and a sequence value, and the messages can include the session key and the sequence value, which are processed through a one-way hash function.

In another embodiment, a computer storage system includes a first computer and a second computer coupled to a communications system. The first computer establishes an authentication handshake with the second computer and periodically sends messages to the second computer. The first computer sends requests to the second computer and the second computer services the requests if the messages are valid and are received within a predetermined time interval. The requests can be to send or receive data, and the authentication handshake can be an exchange of a session key and a sequence value. The

messages can include the session key and the sequence value, which are processed through a one-way hash function.

In another embodiment, the computer storage system includes a first computer and a second computer coupled to a communications system. The first computer establishes an authentication handshake with the second computer and periodically receives messages from the second computer. The first computer receives requests from the second computer and services the requests if the messages are valid and are received within a predetermined time interval. The requests can be to send or receive data, and the authentication handshake can be an exchange of a session key and a sequence value. The messages can include the session key and the sequence value, of which are processed through a one-way hash function.

In another embodiment, a first computer reads a table for information to use in determining expected messages for each of at least two second computers. The table includes identifiers associated with the at least two second computers, session keys associated with the at least two second computers, and sequence values associated with the at least two second computers. The first computer determines the expected messages for each of the at least two second computers, and validates that the expected messages for each of the at least two second computers are identical to each of the corresponding messages from the at least two second computers.

In another embodiment, a computer-executable process is stored on a computer-readable medium. The computer-executable process generates

periodic verification of a computer during requests from the computer to a second computer over a communications system. The computer-executable process includes code to establish an authentication handshake with the second computer, where the authentication handshake can include a session key and a sequence value. There is code to periodically generate and send messages to the second computer, where the messages can include a hashed value of the session key and the sequence value. In another embodiment, the computer-executable process can include code to periodically receive messages and code to service the requests if the messages are valid and are received within a predetermined time interval. The requests can be to send or to receive data.

In another embodiment, the method is carried out in an intelligent storage device for providing periodic verification of a computer during requests from the computer to the intelligent storage device over a communications system. An authentication handshake, which includes a session key and a sequence value, is established with the computer. The intelligent storage device receives messages from the computer and the messages include the session key and the sequence value, which are processed through a one-way hash function. If the messages are valid and are received within a predetermined time interval, the intelligent storage device services the requests. In another embodiment, the method is carried out in a computer for providing periodic verification of the computer during requests from the computer to an intelligent storage device over a communications system. The computer periodically sends messages that include a session key and a sequence value, which are processed through a

one-way hash function. The intelligent storage device services the requests if the messages are valid and are received within a predetermined time interval.

BRIEF DESCRIPTION OF THE DRAWINGS

5 FIG. 1 presents a block diagram of an illustrative arrangement that embodies the principles of the invention; and

 FIG. 2 shows a flow chart of a process carried out in a storage client and a storage server.

10 DETAILED DESCRIPTION

 The present invention improves security of a computer storage system, for example, a storage area network (SAN), by requiring an initiating computer to periodically reaffirm its identity by transmitting a message, called a "heartbeat message," to a servicing computer. The heartbeat message contains a
15 previously established shared secret (for example, a session key) and a sequence value, established by and known only to the original participants. It should be realized that a heartbeat message can include other information. A heartbeat message must be received by the servicing computer within a predetermined time interval in order to maintain data communications between
20 the original participants. It should also be realized that the period at which heartbeat messages are transmitted is independent of data transmissions.

 FIG. 1 presents a general block diagram of an illustrative arrangement that embodies the principles of the invention. It shows a SAN 102, which includes

computers called storage clients 104–110, intelligent storage devices 112, 114, storage devices 116, 118, and computers called storage servers 120, 122.

Collectively, storage device 116 and storage server 122 make up a storage system 124. A communications system 130 interconnects storage clients 104–

5 110, intelligent storage devices 112, 114, storage devices 116, 118, storage servers 120, 122, and storage system 124.

It may be noted that communications system 130 can include any type of network, such as the Internet, local area network (LAN), metropolitan area network (MAN), or wide area network (WAN), so long as the protocols are

10 consistent with the communications protocols utilized by storage clients 104–110, storage devices 112–118, and storage servers 120, 122. Communications system 130 can also comprise various types of networks and topologies, as well as include a high-speed switch for communicating between the storage client computers and the storage server. Communications system 130 may require

15 devices such as hubs, switches and host bus adapters (HBAs) depending on the type of system. Communications system 130 can utilize storage protocols such as Fibre Channel (FC) or Small Computer System Interface (SCSI). The SCSI protocol provides for the interface of personal computers to peripheral hardware, such as disk drives, tape drives, and CD-ROM drives.

20 The data paths can consist of any type of data cable or network used in the transmission of computer data including but not limited to SCSI and Fibre Channel. Fibre Channel is suited for connecting computers to shared storage devices and for interconnecting storage controllers and drives. Because Fibre

Channel was created to transmit large blocks of data very quickly, it is a good transmission interface between computers and clustered storage devices.

The data paths can utilize any data/communications protocol, for example, SCSI or IP, or transmission medium, for example, electrical or optical media,

5 available now or in the future to accomplish computer communications. There is no limitation on the type or format of computer communications, for example packet switched or non-packet switched modes, allowing for short distance, simple cable connection or world-wide, internet connection. As can be observed, the level of flexibility with the above communications system 130 and its
10 communications protocols is almost limitless.

Storage clients 104–110 are computers, for example, personal computers, servers, workstations, or embedded systems, which may need to access data stored at one or more remotely located intelligent storage devices 112, 114, or servers 120, 122. Storage clients 104-110 can run on an operating system such
15 as Microsoft's Windows® Operating System, Unix®, or NetWare®, to name a few. The computers are capable of communicating over the Internet, an intranet, and other networks.

Intelligent storage devices 112, 114 are devices that provide storage or data, and which may include a computer and/or network capabilities. These
20 intelligent storage devices 112, 114 can be, for example, intelligent hard disks, RAID subsystems, intelligent tape drives, and intelligent CD-ROMs, which can be connected to communications system 130. Storage devices 116, 118 can also be

hard disks, RAID subsystems, tape drives, and CD-ROMs included in storage server 120 or part of storage system 124.

Storage servers 120, 122 are computers, for example, personal computers, servers, workstations, or embedded systems, which are coupled to one or more storage devices, such as storage devices 116 or 118. These storage devices 116, 118 are situated so that the data communication paths from the storage devices pass through the corresponding storage servers 120, 122, before connecting to one or more storage clients 104-110.

A computer is any device that accepts information (in the form of digital data) and manipulates it for some result based on a program or sequence of instructions.

The present invention is embodied in software programs that provide improved security by requiring an initiating storage client to periodically reaffirm its identity by transmitting an additional message, called a heartbeat message, to a storage server. The software programs check for unauthorized access of storage devices 112-118 and provide periodic verification of storage clients 104-110 during data transfer (i.e., reading data and/or writing data) between storage clients 104-110 and storage devices 112-118, while not decreasing data throughput. Generally, there are two types of software programs in the present invention. The first type, called a "storage security agent" (SSAgent), is stored in storage clients 104-110, and the second type called a "storage security administrator" (SSAdmin) is stored in intelligent storage devices 112, 114 and storage servers 120-122. These software programs comprise computer-

executable code (i.e., processing steps) stored in a computer readable medium, which includes any kind of computer memory such as floppy disks, hard disks, CD-ROMs, flash ROMs, nonvolatile ROM and RAM.

To illustrate, the SSAdmin is generally stored in a storage server or intelligent storage device, for example, storage server 122, and the SSAgent is stored in a storage client, for example, storage client 104. Generally, the SSAgent has two responsibilities, the first of which is to authenticate storage client 104 with storage server 122 and the second of which is to periodically provide a heartbeat signal. If the authentication is approved by storage server 122, then storage client 104 can start sending requests (e.g., reading data and/or writing data) to storage server 122 and storage server 122 can start servicing the requests.

The first responsibility, authentication, is a process in which an entity, such as storage client 104, must identify itself to a system, such as storage server 122, before services are provided to storage client 104. This process protects storage server 122 against unauthorized access and against counterfeit computers that may try to copy the identity of a real storage client in order to gain access to storage server 122. Key exchanges or passwords are commonly used to authenticate users.

The second responsibility of the SSAgent is to periodically provide a heartbeat message, while storage client 104 sends requests to storage server 122 and storage server 122 services the requests. The heartbeat message is provided periodically to avoid decreasing data throughput. The heartbeat

message includes a predetermined sequence value and is sent within a predetermined time interval, to inform storage server 122 that storage client 104 is the storage client that storage server 122 authenticated, not an unauthorized computer. The sequence value can be any value comprising numbers and/or letters and is updated by the SSAgent in storage client 104 and by the SSAdmin in storage server 122 after each heartbeat message is sent. One way to update the sequence value is to increment it by a predetermined increment value. The predetermined time interval is a time, for example, 5 seconds, in which a heartbeat message must be sent and received. If a valid heartbeat message is sent and received within the predetermined time interval, then the time in which to receive the next heartbeat message is extended by a time equal to the predetermined time interval, which in this example is 5 seconds. Both the SSAdmin and the SSAgent maintain synchronized timers that are used to determine whether the heartbeat messages are received within the predetermined time intervals. The predetermined sequence value and time interval are used to distinguish the real storage client from a counterfeit storage client.

The heartbeat message is part of a monitoring process in which a computer, such as storage server 122, must periodically receive a heartbeat message from an entity, such as storage client 104, to indicate that storage client 104 is present and to continue to allow the requests sent by storage client 104 to be serviced. If storage server 122 does not receive a heartbeat message within the predetermined time interval, requests (e.g., requests to send (write) or

receive (read) data) from storage client 104 will no longer be accepted, until the next authentication. This process protects storage server 122 from sending data out onto communications system 130 and to a masquerading computer, in the event that storage client 104 is no longer functional or becomes disconnected.

5. Example requests are read or write commands to storage server 122 to read data from or write data to the storage device 116.

Continuing with the illustrative example, the SSAdmin receives authentication and heartbeat messages from the SSAdmin in storage client 104. The SSAdmin negotiates the authentication process and manages the monitoring of the heartbeat messages from all of the storage clients which would like to utilize storage server 122. The SSAdmin maintains a list of all the valid storage clients and their security information. It should be realized that the SSAdmin can maintain and service multiple storage clients.

The SSAdmin instructs storage server 122 to continue to service requests from storage client 104 for as long as valid heartbeat messages are received within the predetermined time interval. Service will be halted when storage client 104 either intentionally ends its connection or when a valid heartbeat message fails to be received by storage server 122 within the predetermined time interval. If an invalid or unknown heartbeat message is received by storage server 122, the invalid or unknown heartbeat message is ignored. When the predetermined time interval expires, the time to receive the next message is not extended and the SSAdmin instructs storage server 122 to discontinue servicing requests from storage client 104. If, in the case above, a valid heartbeat message is sent after

the invalid or unknown heartbeat message, but before the predetermined time interval expires, then the SSAdmin instructs storage server 122 to continue servicing requests from storage client 104, and the time to receive the next heartbeat message is extended by a time equal to the predetermined time interval. If a valid heartbeat message is received within the predetermined time interval, the SSAdmin instructs storage server 122 to continue servicing requests from storage client 104, and the time to receive the next heartbeat message is extended by a time equal to the predetermined time interval. If a valid heartbeat message is not received within the predetermined time interval, then the time to receive the next heartbeat message is not extended and the SSAdmin instructs storage server 122 to discontinue servicing requests from storage client 104. If an invalid or unknown heartbeat message is received by storage server 122, and another valid heartbeat message is not received within the predetermined time interval, then the SSAdmin instructs storage server 122 to discontinue servicing requests from storage client 104. Only a valid message within the predetermined time interval can trigger the SSAdmin to extend the timer to the next time interval, and therefore maintain servicing requests.

Operationally, before the SSAgent and the SSAdmin engage in an authentication process, they need preestablished authentication information. A shared secret is established in both the SSAdmin and the SSAgent. Using processes and techniques known within the art of cryptography, a shared secret can be distributed to the SSAgent and SSAdmin either manually by a person or through a key exchange protocol, such as Diffie-Hellman.

The shared secret is securely stored in storage client 104 and storage server 122 in a manner to prevent and discourage individuals from determining the shared secret's identity. Each storage client, for example storage clients 104-110, is given its own individual shared secret to enhance the security level of storage area network 102.

Continuing the illustration, an authentication handshake process begins by establishing a secure session between the SSAdmin in storage client 104 and SSAdmin in storage server 122. The SSAdmin instructs storage client 104 to perform a standard challenge-response protocol using the shared secret with the SSAdmin. During this process a session key is generated, a sequence value is determined, an increment value is determined, and an identifier (e.g., IP address) associated with storage client 104 is stored in a table in memory. This information is stored in a table for the purpose of maintaining the session keys, sequence values, and increment values for multiple storage clients 104-110. It should be noted that the increment value can be predetermined and embedded as part of the SSAdmin and SSAdmin software. In this illustration, the sequence value is randomly generated by SSAdmin and shared with SSAdmin during the authentication process. It should be realized that one skilled in the art could arrange for the sequence value to be generated by SSAdmin. The session key and the sequence value, along with any other necessary information will be used in subsequent heartbeat messages, which are described in more detail below. Once the SSAdmin validates the identity of storage client 104 by the authentication process, the SSAdmin instructs the storage server 122 to service

requests from storage client 104. It should be realized that other storage clients and/or multiple storage clients may be used, such as storage clients 106-110 or more, and that they can interact with various storage devices, such as intelligent storage devices 112, 114, and/or storage device 118 via storage server 120.

5 FIG. 2 illustrates an exemplary heartbeat process using, for example, storage server 122 and storage client 104. At block 202 the SSAgent instructs storage client 104 to retrieve the session key and the agreed upon sequence value for use in creating a heartbeat message. SSAdmin instructs storage server 122 to update (e.g., by a predetermined increment value) the sequence value to
10 a new value after it is determined that a valid heartbeat message was received and to send an acknowledgment message to the SSAgent in storage client 104. Upon receipt of the acknowledgment message SSAgent instructs storage client 104 to update its sequence value. Because heartbeat messages are unique to every storage client, the identity of storage client 104 is revalidated after each
15 heartbeat message is received. The sequence values are known only between storage server 122 and storage client 104.

At block 204, the SSAgent instructs storage client 104 to process the session key and the sequence value through a one-way hash function (algorithm) to generate a hash value of the heartbeat message that is going to be sent to the
20 SSAdmin in storage server 122. It should be realized that the information that comprises a heartbeat message can be any type of information so long as the information satisfies security and uniqueness requirements. In this illustration, the one-way hash algorithm is preferably Message-Digest Hash Function 5 (MD5),

but other algorithms, such as Secure Hash Algorithm (SHA-1), can be used. MD5 is a digital signature algorithm that is used to verify data integrity through the creation of a 128-bit message digest from data input, which may be a message of any length.

5 The SSAdmin, at block 206, instructs storage server 122 to retrieve the session key and sequence value, which it expects to receive from storage client 104. To determine which session key, sequence value, and increment value to retrieve, the SSAdmin receives an identifier (e.g., IP address) from the SSAgent, searches the table for a similar identifier, and associates the identifier with the
10 session key and sequence value established during authentication. The SSAdmin instructs storage server 122, at block 208, to process the session key and sequence value through a one-way hash algorithm to generate a hashed value of an expected heartbeat message. If the storage client 104 is the client that was authenticated earlier, then the expected heartbeat message should be
15 the same as the heartbeat message generated by storage client 104 at block 204. It should be realized that blocks 202-204 and blocks 206-208 can be performed as parallel operations.

 Alternatively, SSAdmin can instruct storage server 122 to process the session key and sequence value for each storage client 104-110 through a one-
20 way hash algorithm and store the results (expected heartbeat message) in the table. Thus, when comparing the expected heartbeat message to the received heartbeat message, SSAdmin can instruct storage server 122 to read the

identifier and search the table for the expected heartbeat message, before comparing the messages to each other.

While storage client 104 sends requests to storage server 122 and storage server 122 services the requests, the SSAdmin, at block 210, instructs storage client 104 to send the heartbeat message to storage server 122 over communications system 130. Storage server 122 receives the heartbeat message from storage client 104 and, at block 212, SSAdmin instructs storage server 122 to compare the heartbeat message from storage client 104 to the expected heartbeat message generated at block 210. If the heartbeat messages are not the same they are considered invalid or unknown and, at block 214, SSAdmin instructs storage server 122 to continue servicing requests. If, at block 222, a valid heartbeat message is not received within the predetermined time interval, the SSAdmin instructs storage server 122, at block 224, to discontinue servicing any more requests from the SSAdmin at storage client 104. If a valid message is received within the predetermined time interval, the SSAdmin instructs storage server 122 to continue servicing requests, at block 220. Further, SSAdmin can instruct storage server 122 to generate and send a message to storage client 104 requesting that the heartbeat message be resent. SSAdmin can also instruct storage server 122 to generate and send a warning signal, such as "Possible Intruder," to the display of storage server 122 if invalid or unknown heartbeat messages are received. It should be realized that the amount of data that may be sent to a masquerading computer or unauthorized computer depends upon how quickly storage server 122 detects the masquerading or

unauthorized computer, which further depends upon the time interval within which heartbeat messages are sent. Using a short time interval will result in quicker detection than using a long time interval.

If, at block 212, the heartbeat messages are the same, the SSAdmin
5 instructs storage server 122, at block 216, to determine whether the heartbeat message from storage client 104 was received by storage server 122 within a predetermined time interval. If the heartbeat message was not received within the predetermined time interval, at block 218, SSAdmin instructs storage server 122 to stop servicing requests from storage client 104. If the heartbeat message
10 was received within the predetermined time interval, the SSAdmin, at block 220, instructs storage server 122 to continue servicing the requests from storage client 104.

The above presents various principles and features of the invention through descriptions of various embodiments. It is understood that skilled
15 artisans can make various changes and modifications to the embodiments without departing from the spirit and scope of this invention. For example, one of ordinary skill in the art would recognize that, although the invention has been described by reference to a client-server relationship, an alternative embodiment of the invention can utilize a peer-to-peer relationship. Thus, one of ordinary skill
20 in the art would understand that the invention is not to be limited by the foregoing illustrative details, but rather is to be defined by the following claims.